by GERALD COHN

Experiment with digital counters

Like to learn more about digital counters and how they work? The best way to do this is to experiment with a low-cost counter IC, the 7490. You can run it through its paces with the simple square wave oscillator we described in the August issue.

by GERALD COHN

The counter that we are going to build is of the decade type, the name decade being derived from the Latin word "decem" which means ten. This means that the counter is capable of counting to 10 before the whole cycle is repeated.

All that we need for the experiment is an easy-to-get 7490 digital counter IC and a handful of other components. The actual "workings" inside the IC consist of four master-slave flip-flops and some additional gating, providing the reset and count control functions. A brief description of flipflop and basic counter operation is provided in the box.

The 7490 is a decade counter device, which counts in a modified binary fashion known as "BCD". There are several ways in which binary numbers can be used in digital counting, one of which is known as BCD or "binary coded decimal". Broadly speaking a code of this sort must use four binary digits or "bits" to represent each decimal digit. This is because this many bits must be used to provide 10 different value combinations. Four bits actually provide 16 different combinations, but the remaining six combinations are unused.

For those readers that are not so familiar with the binary numbering system, we will go over some of the basic rules of the system. You will find that these are going to prove quite important later on, in particular when you have completed the construction of the project and are ready to use it.

To handle decimal numbers in digital circuits is rather cumbersome, and also demands greater circuit complexity. An alternative is to change the format of the numbers into what is known as "binary". Binary notation uses 2 as its base, in place of the 10 that is used in the decimal system. Binary notation is very well suited to digital circuits, as it involves only two numeral values: 0 and 1. These replace the 10 numeral values (0, 1, 2, . . . 9) used in decimal notation.

This means that many more digits must be used to represent a given number, because each numeral digit position represents a certain power of 2 rather than a power of 10. In place of the units-tens-hundreds-thousands progression of the decimal system, the numeral positions represent units-twos-fours-eights and so on.

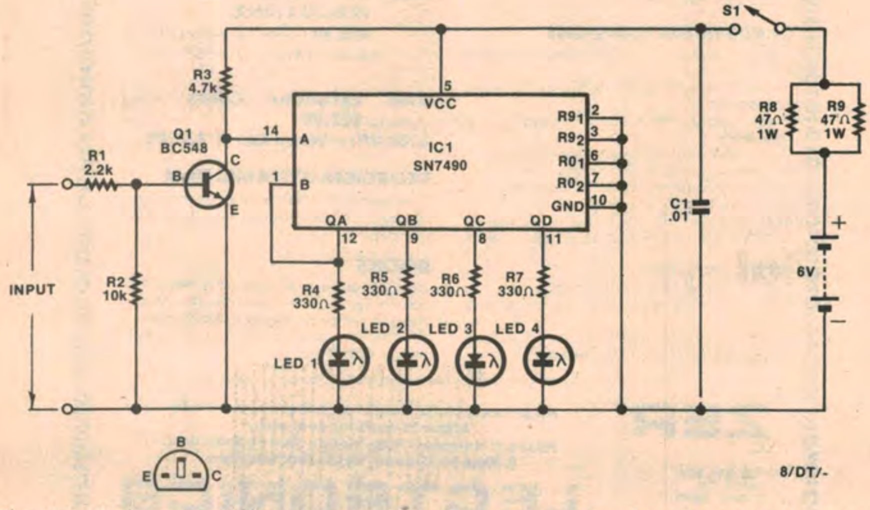
In this sense the binary system is clumsier. But each numeral position has only two values, so that it may be represented very simply by a circuit voltage or current switching between two levels — high and low.

Hooking up the 7490 as an experimental counter is a simple task, and if performed with care should only take about two hours. You will probably find that apart from the IC and the LEDs, all the other components will be in your junk box.

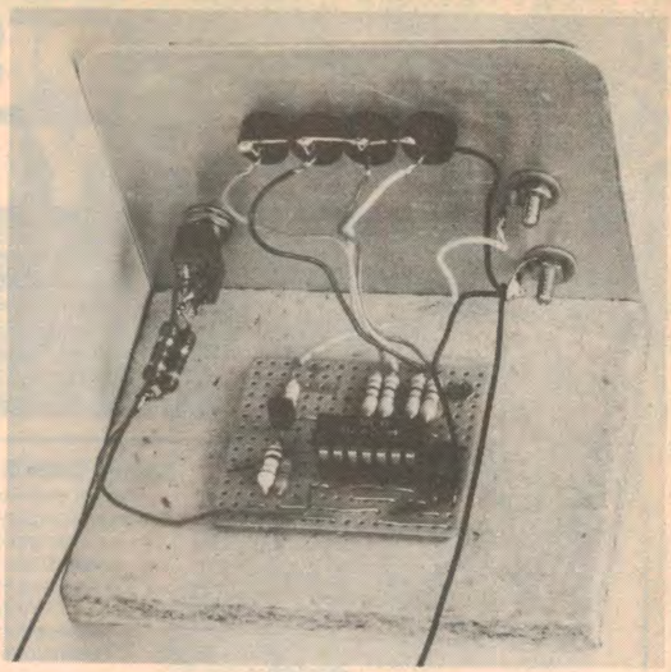
The components are all mounted on a small piece of Vero-board, keeping the cost to a minimum. The input stage to the counter consists of an NPN transistor inverter that performs the function of level translation. The reason that level translation is required is that the oscillator used to provide the input pulses uses a 9V supply, whereas the counter circuit operates from a 5V supply.

The 7490 IC that is used in this counter is of the TTL type, and this type of logic is designed to operate from a 5V supply rail. The level translator at the input to the counter will prove to be quite useful, since it will enable the counter to be used with all sorts of equipment operating at various different supply voltages.

The output of the transistor inverter is fed to the input of the counter IC, which is at pin 14. The counter IC features two inputs because it is internally made up of two separate counters, the first of which is a divide by 2 counter and the second being a divide by 5 counter. The output of the



The circuit diagram of the counter. Note the inverting level translator (Q1) at the A input to the IC.



Photographs showing the construction of the unit. The front panel layout appears above, while the rear view of the unit is shown to the right.

divide by 2 counter is fed to the input of the divide by five counter, this being done via a link between pins 1 and 12 of the 7490 IC.

The counter IC also has four other inputs that perform the functions of resetting the counter to either 0 or 9. Since we are not interested in using these reset functions, we ground the inputs. The output of the counter takes the form of four LEDS (light emitting diodes), arranged in such a way that they provide a binary readout. The output can be easily read if you consider a lighted LED to indicate a logic 1, and an unlit LED to indicate a logic 0.

As you can see from the circuit diagram, each of the LEDs has a resistor in series with it. The reason for the series resistor is to limit the amount of current that passes through the LED to approximately 10 miliamps. This is plenty of current to ensure that the LEDS have adequate levels of light output.

Before you start placing components onto the strip board, cut the appropriate tracks as shown in the component overlay diagram. To cut the tracks is a simple matter requiring the use of a small twist drill. The tip of the drill is placed in the hole where the copper track is to be cut and then rotated until the drill has cut away the copper that is not required. The finished result will be a small countersink in the board, with the copper strip milled away.

Once you have cut the tracks that are indicated on the overlay, start to mount the components, again following the layout shown by the component overlay. The first components that should be mounted onto the board are the resistors and the capacitor. These can then be followed up by the transistor. The IC is the last component that is to be mounted, and we suggest that an IC socket be used for this, therefore enabling the IC to be used again in a future project without any damage resulting from soldering and then removing.

Once the board has been assembled, we turn our attention to the mechanical construction of the project. As can be seen from the photograph, we mounted the prototype onto a piece of particle board, and mounted the LEDs in a front panel that was made from a small piece of aluminium sheet. Also mounted on the front panel are the terminals for the input to the counter, and the power switch. The two 47ohm 1W resistors are mounted on the rear of the power switch, and their purpose is to drop the battery voltage from 6V to approximately 5V, and at the same time limit the amount of current that the circuit can draw. The maximum current drain on the battery will occur when the counter reaches the count of 7, since three of the four LEDs will be lit up. Since this is a BCD counter, this will be the maximum number of LEDs that will be lit up at any one time.

All the holes in the front panel were drilled to a size of 6.5mm, and these are for the mounting of the LEDs, the power switch and the terminal posts. The wiring to the LEDs should be done carefully to ensure that the order of the outputs is correct. The cathodes of the LEDs are wired up to a common bus and then taken to ground, together with the black terminal post. The input of the counter is taken to the red terminal post.

We used a 6V lantern battery for the prototype. It is not essential that this type of battery be used, but we found it to be the best since it is a heavy duty battery. You could also use a battery clip that contains four type AA batteries to obtain a six volt source.

We estimate that the current cost of the parts for this project is approximately

$6.00

This includes sales tax.

Led lead identification

The cathode is the shorter of the two leads or can be identified by a flat edge on the plastic body.

THE PARTS LIST

1 7490 TTL IC
1 BC548 NPN transistor
2 47 ohm ½W resistors
4 330 ohm resistors
1 2.2k ohm resistor
1 4.7k ohm resistor
1 10k ohm resistor
1 .01uF polyester capacitor
4 Red LEDs and bezels to suit
1 Piece Vero board, 44 x 40mm

MISCELLANEOUS

Piece of aluminium sheet 7cm x 9cm, piece of particle board 7cm x 9cm, miniature single pole toggle switch, terminal posts (1 red and 1 black), IC socket (14 pin), solder tags, solder, hookup wire etc.

FLIPFLOPS: what they are & how they work.

Apart from logic gates, probably the most basic elements in digital circuits are flipflops. As the name suggests, a flipflop is a bistable device. It has two stable operating conditions, and can be made to switch from one to the other.

A very basic flipflop can be formed by connecting two 2-input NAND gates, as shown in Fig. 1. As you can see, the two gates are "cross coupled", with the output of each being connected to one of the inputs of the other. This forms a regenerative feedback loop, with the result that if the two uncommitted inputs are taken to true (1) logic level, only one of the gates can have a true output; the other must have a false (0) output.

This happens because the output that is true causes both inputs of the other gate to be true, driving the second gate's output false by virtue of the inherent inversion in a NAND element.

Only one output of the flipflop (or "FF") is normally true, and the other is false. The two outputs are logically complementary, in other words — each is the logical complement of the other. By convention one output is usually labelled "Q" and the other "Q-bar".

The FF has two stable states then: one with Q true and Q-bar false, and the other with Q false and Q-bar true. Just which of the two states the FF is in depends upon its previous history.

The main factor that determines the state of the flipflop is any signals which may be applied to the inputs A and B. There are various possibilities here: the FF may have either Q or Q-bar true initially, while either A or B may be taken true or false independently. With three variables, this gives eight different situations, each of which can be analysed using basic gate principles.

The results are shown in concise form in the "truth table" of Fig. 1. Each pair of lines covers one of the four possible true-value combinations for the two inputs, with the other two lines in each pair covering the two possible initial states of the flipflop.

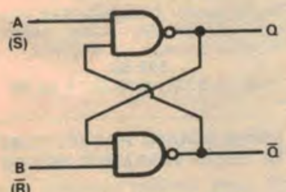
A simple flipflop of the type in Fig. 1 is known as a "Reset-Set" or R-S flipflop. While an RS flipflop has its uses, there are many applications in digital systems that require a FF to respond only at certain fixed times, as determined by general timing or "clock" signals that are fed throughout the system. To provide for this sort of operation, a number of variations on the basic flipflop have been evolved.

One of these is the so-called T-type or "toggle" flipflop, shown in Fig. 2. This has two inputs, like the RS type, but here the inputs perform entirely different functions. One input is called the toggle or "T" input and the other the "clock" input.

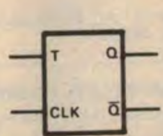
Referring to the truth table of Fig. 2, you can see that if the T input is held false (0 logic level) during the clock pulse, this has the effect of "freezing" the FF in its initial state; ie, no change occurs at the outputs. However if the T input is held true (logic 1) during the clock pulse, the clock pulse forces the FF to "toggle" or change states — regardless of its initial state. If it was set, it will reset; and vice-versa.

The T-type FF is thus capable of only two responses to a clock pulse. It can either remain unchanged, or toggle, depending upon the logic level applied to the T input.

Probably the most important type of clocked flipflop is the JK type, which is shown in Fig. 3. This is a very flexible element (and also the type that is used in the counter IC around which this project is built), in that it can be arranged to perform all of the functions performed by the other types.



A	B	BEFORE		AFTER	
		Q	$\bar{Q}$	Q	$\bar{Q}$
1	1	0	1	0	1
1	1	1	0	1	0
0	1	0	1	1	0
0	1	1	0	1	0
1	0	0	1	0	1
1	0	1	0	0	1
0	0	0	1	?	?
0	0	1	0	?	?



T	BEFORE CLK PULSE		AFTER CLK PULSE	
	Q	$\bar{Q}$	Q	$\bar{Q}$
0	0	1	0	1
0	1	0	1	0
1	0	1	1	0
1	1	0	0	1

Fig. 1 to the left shows the SR flipflop, while Fig. 2 above shows the T-type flipflop and its truth table.

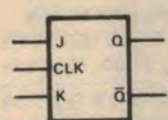


Fig. 3 The most widely used of the flipflop family, the JK type, shown here with its truth table.

J	K	Q (BEFORE)	Q (AFTER)
0	0	0	0
0	0	1	1
1	0	0	1
1	0	1	1
0	1	0	0
0	1	1	0
1	1	0	1
1	1	1	0

The counter could also be powered from a power supply that runs from the mains, and we will consider publishing a power supply project of this type sometime in the not too distant future.

Now that you have completed the assembly of the project, go through and check all the wiring once more, and don't forget the board assembly. If you are satisfied that all is in order, hook up the square wave oscillator that was featured in the August issue, and turn on the power to the counter. Now turn the switch on the oscillator into the 1Hz position and note that the LEDs of the counter's display should start to follow the binary counting sequence. To check that the sequence does in fact follow the binary notation, refer back to table 1 and check that the sequence of the LEDs corresponds to that shown in the table.

The counter does have some other more practical applications other than flashing a row of lights in a binary sequence. You could for example use it as a frequency divider. If you feed a frequency into the input, you can obtain two different output frequencies; one corresponding to the input divided by two, and the other to the input divided by 10. So if we feed a frequency of 1000Hz into the input, the resulting output frequencies will be 500Hz (1000/z) at output A and 100Hz (1000/10) at output D.

As you can see from the above information, the input frequency is divided down by a factor of 2 each time, and the different output frequencies appear at the outputs of the counter IC.

If you did build the oscillator project, you will soon discover that you can obtain 25 different frequencies, if you use the oscillator together with this counter project.

For those experimenters that feel a little more adventurous, you could

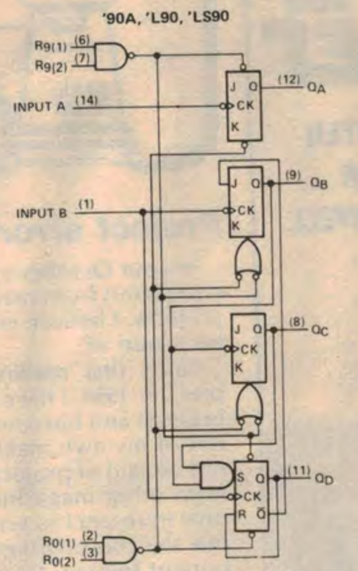
Flipflops in counters

The JK flipflop has two main inputs in addition to the clock input, and as you can see these are labelled "J" and "K". If the J input alone is held true, the FF will set; if the K input alone is held true, the FF will reset; if both the J and the K inputs are held true together, the flipflop will toggle with the applied clock; and if both are held false, the flipflop will "freeze".

If you now refer to the diagram of Fig. 4, you will see that we have reproduced the circuit of the 7490 IC. Note that it uses JK type flipflops.

Closer examination of the schematic will reveal that the first FF in the counter chain is in fact not connected to the other FFs. The device is actually a dual counter in that the first stage is a divide-by-2 stage, while the other three FFs form a divide-by-5 stage. To use this device as a divide-by-10 stage, we must connect the two internal stages in cascade. This is done by connecting the output of the first stage to the input of the second stage (ie pins 12 and 1 are connected together).

The device also features some special reset inputs, but we shall leave these from the discussion as they are not used in this project. The pulse train that is to be counted is fed to the input of the first stage, ie. pin 14. The output from the device is in a four bit binary pattern, actually not pure binary, rather BCD or "binary coded decimal". The outputs are labelled QA to QD, with QA being the least significant bit (LSB), and QD being the most significant bit (MSB).



COUNT	OUTPUT			
	Q_D	Q_C	Q_B	Q_A
0	L	L	L	L
1	L	L	L	H
2	L	L	H	L
3	L	L	H	H
4	L	H	L	L
5	L	H	L	H
6	L	H	H	L
7	L	H	H	H
8	H	L	L	L
9	H	L	L	H

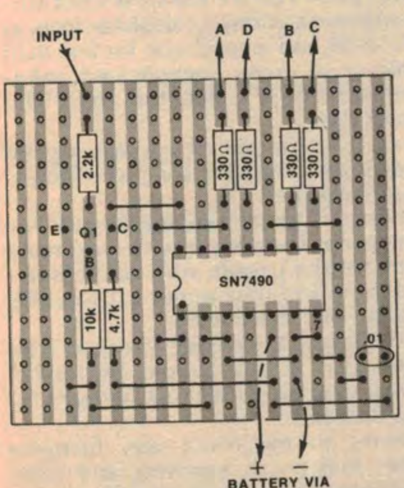
Fig. 4 to the right shows the circuit of the 7490 IC. Two separate counters exist in the package, the first a divide by two, and the second a divide by five. Fig. 5 above shows the truth table for the outputs of the 7490 counter.

The resulting output from the counter will appear as shown in Fig. 5. Here you can see the BCD code that is output, and also the decimal equivalent of the BCD number.

For those readers that wish to learn a little more about flip-flops and electronic counters we refer you to the Electronics Australia handbook "An Introduction To Digital Electronics".

build up several of these counters and interconnect them to obtain counts that are increasing in decade steps. In other words, if you use two of these counters, you can count up to 100, and with three you can count up to 1000, and so on. To connect one counter to another, all that has to be done is to take the output of the first counter (output labelled QD), and feed this to the input of the next, and so on.

By cascading counters in this way you can obtain count lengths that are as long as you could want them. All that is required is to build several of these, one for each decade. If for example you want to be able to count to one million, you will need to build six of these counter modules and then interconnect them the way that was described in the last paragraph. The number of applications for a counter of this type is almost enedless. All that you need to do is come up with some more ideas, other than those that have already been described here.



The component overlay diagram. Note the cuts in the copper tracks.

In an early issue we hope to show you how to hook up a digital display to your counter, so you can read the count in both binary and decimal form. Apart from being an extremely practical and fun project, it also goes a long way in teaching you some of the basics that are involved in electronic counting.

Incidentally, a little message to all you beginners out there. This is your section of the magazine, and we would welcome suggestions for projects and ideas that you would like to see designed and published. You are the people that know what you want, and you could greatly simplify our task in project selection if you let us know.

In the meantime, we hope that you have lots of fun with the counter, and if you have any novel ideas for the use of the counter, let us know so that we can pass some of these along to other readers.